Software Release 2.3.1
For Rapier Switches, AR300 and AR700 Series Routers, and AR800 Series Modular Switching Routers

Allied Telesyn
Simply connecting the world

Introduction

Allied Telesyn International announces the release of Software Release 2.3.1 on the AR300 and AR700 Series routers, Rapier Series layer 3 switches, and AR800 Series modular switching routers. This release note describes software features that are new since Software Release 2.2.2. It should be read in conjunction with the Quick Install Guide, Quick Start Guide, User Guide, Hardware Reference and Software Reference for your router or switch. These documents can be found on the Documentation and Tools CD-ROM packaged with your router or switch, or on the support site at:
www.alliedtelesyn.co.nz/documentation/documentation.html

The main new features in release 2.3.1 are:

■ Border Gateway Protocol Phase 1a

■ Paladin Firewall HTTP Proxy (Application Gateway), additional Firewall NAT features and rule expiry

■ Support for Rapier i Series layer three switches.

WARNING: Information in this release note is subject to change without notice and does not represent a commitment on the part of Allied Telesyn International. While every effort has been made to ensure that the information contained within this document and the features and changes described are accurate, Allied Telesyn International can not accept any type of liability for errors in, or omissions arising from the use of this information.

Hardware Platforms

Software Release 2.3.1 is available for the following hardware platforms:

■ AR300 Series Routers

■ AR700 Series Routers

■ AR800 Series Modular Switching Routers

■ Rapier Series Layer 3 Switches

Software Release 2.3.1 will support the Rapier i Series hardware platforms as they become available ("*Rapier i Series*" on page 2).

For existing models with Network Service Module (NSM) bays, Software Release 2.3.1 supports hot swapping of NSMs, and some PICs in those NSMs, so that they can be installed and uninstalled without the need to power down the entire router or switch ("*Hot Swapping Network Service Modules*" on page 3).

Rapier i Series

The Rapier i Series layer 3 switches will provide all the features of the original Rapier series. While the first software release on these hardware models will provide the same features as the original Rapier Series (plus bandwidth limiting), the hardware on the Rapier i Series layer 3 switches will allow later software releases to provide enhanced Virtual LAN and Quality of Service features.

Hot Swapping Network Service Modules

In routers and switches that have NSM bays, this release allows the following NSMs to be hot swapped, so that they can be installed and uninstalled without powering down the entire router or switch:

■ AT-AR040 NSM with 4 PIC slots (NSM-4PIC)

■ AT-AR041 NSM with 8 BRI S/T WAN ports (NSM-8BRI)

■ AT-AR042 NSM with 4 BRI S/T WAN Ports (NSM-4BRI)

The following PIC cards can be hot swapped if they are in NSM bays:

■ AT-AR021(S) PIC BRI (S)

■ AT-AR021(U) PIC BRI (U)

■ AT-AR022 PIC Eth

■ AT-AR023 PIC Sync

■ AT-AR026 PIC 10/100 Eth

PICs in PIC bays in base router units (for instance, the AR720 and AR740 routers) do not support hot swapping. The PICs in an NSM can only be hot swapped by preparing the NSM bay for hotswap.

An NSM, with or without PICs, can be hot inserted into a previously empty bay. Hot inserted cards behave as though they had been present at router start-up, except that the router configuration script will not be scanned for commands that may relate to interfaces on the hot-inserted cards.

An NSM, with or without PIC cards, can be hot swapped out, and an identical combination of NSM and PIC cards can be hot swapped into the same bay. The software configurations of the interfaces on the hot-swapped cards are preserved across the hot swap so that modules configured to interfaces on the cards can continue to use the interfaces.

An NSM, with or without PICs, can be hot swapped out and a different combination of NSM and PICs can be hot swapped into the same bay. For any card in the combination that is replaced by a card of a different type, software interface instances for the old card are destroyed and their configurations forgotten, and new interface instances are created from scratch for the new card. For any card in the combination that is replaced by a card of the same type, interface instances are preserved.

NSM-4PIC (AR040) only: Cards of the same type but with differing manufacturing revision levels may in some cases be treated as cards of different types when hot swapping.

WARNING: It is important to observe the following procedure carefully when hot swapping NSMs. Failure to follow this procedure will cause the router to crash, and may cause damage to files stored in FLASH.
Do not attempt to hot swap while the contents of FLASH memory are being modified, for instance when files are being loaded onto the router or during FLASH compaction. Hot swapping while FLASH memory is being modified may corrupt FLASH memory, damaging configuration files, software release

files, feature licences and other files. (If this happens, FLASH memory may need to be cleared completely, leaving no functioning software to run the router.)
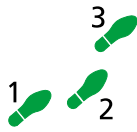
Hot swap an NSM out of an NSM bay

Follow these steps to hot swap an NSM, or PICs in an NSM-4PIC, out of an NSM bay.

1. **Prepare the NSM bay for hot swap.**

    Look at the "**Swap**" and "**In use**" LEDs beside the NSM bay. If the "**In Use**" LED is lit, press the "**Hot Swap**" switch slowly using a pointed object such as a pencil tip. The "**In Use**" LED should go out and the "**Swap**" LED should light.

    If the "**In Use**" LED remains lit, or if neither of the LEDs beside the NSM bay is lit, the router software release does not support hot swap, and the router must be switched off to remove the NSM.

2. **Remove the NSM or PIC.**

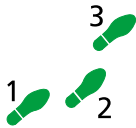   When the "**Swap**" LED is lit, remove the NSM or the PIC that is being swapped.

Hot swap an NSM into an NSM bay

Follow these steps to hot swap an NSM, or PICs in an NSM-4PIC, into an empty NSM bay.

1. **Check that the NSM or PIC bay is empty.**

2. **Check that the NSM bay is ready for hot swap.**

    Look at the "**Swap**" and "**In use**" LEDs beside the NSM bay. The "**Swap**" LED should be lit.

    If the "**In Use**" LED is lit, press the "**Hot Swap**" switch slowly using a pointed object such as a pencil tip. The "**In Use**" LED should go out and the "**Swap**" LED should light.

    If the "**In Use**" LED remains lit, or if neither of the LEDs beside the NSM bay is lit, the router software release does not support hot swap, and the router must be switched off to remove the NSM.

3. **Insert the NSM or PIC.**

    When the "Swap" LED is lit, insert the NSM or PIC.

4. **Return the NSM bay to use.**

    Press the "**Hot Swap**" switch using a pointed object such as a pencil tip. The "**Swap**" LED will go out and the "**In Use**" LED will light.

    If the "**In Use**" LED stays lit only briefly then the "**Swap**" LED lights again, the NSM is of a type that the software release does not support.

For information about the behaviour of interfaces during and after NSM hot swapping, see "*NSM Hot Swap Software Support*" on page 6.

Software Features

The following features are available on all routers and switches supported by this release, unless otherwise stated:

Major features

■ NSM Hot Swap software support for models with NSM bays ("*NSM Hot Swap Software Support*" on page 6)

■ Domain Name Server Enhancements (IP) ("*Domain Name Server Enhancements*" on page 7)

■ Configurable Telnet Server Port Number ("*Telnet Server Port Number*" on page 9)

■ Up and down triggers for Ethernet interfaces ("*Triggers for Ethernet Interfaces*" on page 9)

■ Changes to the number of encryption and compression channels, depending on the amount of RAM on the router or switch ("*ENCO Channels*" on page 10)

■ IP Security (IPsec) enhancements: the Source Interface can be now be specified, and IPsec performance is enhanced ("*IP Security (IPsec) Source Interface and Enhancements*" on page 11)

■ OSPF on Demand ("*OSPF on Demand*" on page 12)

■ Paladin Firewall Enhancements ("*Paladin Firewall Enhancements*" on page 14)

■ Paladin Firewall HTTP Application Gateway (Proxy) ("*Paladin Firewall HTTP Application Gateway (Proxy)*" on page 21)

■ VRRP Port Monitoring ("*VRRP Port Monitoring*" on page 26)

■ Border Gateway Protocol version 4, phase 1 ("*Border Gateway Protocol 4 (BGP-4)*" on page 28) (not available on AR300 Series routers).

■ Commands to reset interface and IP MIB counters to zero, and changes to the display of MIB counters ("*IP and Interface Counters*" on page 29)

■ An extended range of telephony functions, on AR300 and AR310 routers ("*Telephony (PBX) Functionality*" on page 33)

■ Bandwidth limiting on Rapier i Series switch ports ("*Bandwidth Limiting*" on page 34)

Minor improvements

■ The DHCP server is now able to successfully allocate addresses to Macintosh devices running Open Transport version 2.5.1 and 2.5.2.

■ To increase switch security, the INFILTERING parameter of the SET SWITCH PORT command now defaults to ON.

The INFILTERING parameter enables or disables Ingress Filtering of frames admitted according to the ACCEPTABLE parameter, on the specified ports. Each port on the switch belongs to one or more VLANs. If INFILTERING is set to ON, Ingress Filtering is enabled: any frame received on a specified port is only admitted if the port belongs to the VLAN with which the frame is associated. Conversely, any frame received on the port is discarded if the port does not belong to the VLAN with which the frame is associated. Untagged frames admitted by the ACCEPTABLE parameter are admitted, since they have the numerical VLAN Identifier (VID) of the VLAN for which the port in an untagged member. If OFF is specified, Ingress Filtering is disabled, and no frames are discarded by this part of the Ingress Rules.

This change does not apply to AR300 or AR700 Series routers.

NSM Hot Swap Software Support

When a card is hot-swapped out of a bay, its interface instances become dormant. They stay dormant until either another card of the same type is hot-swapped into the bay, in which case they are reactivated, or a card of a different type is hot-swapped into the bay, in which case they are destroyed.

Dormant interfaces are included in the SHOW INTERFACE command output and in the SNMP interfaces MIB, marked as swapped out. In other router or switch commands, however, the router or switch behaves as though dormant interfaces do not exist.

Instances of higher-level modules such as LAPD and Q931, ISDNCC, PPP, and IP, that are attached to an interface that becomes dormant, do not themselves become dormant. They behave as if the interface has stopped communicating, for example as if its cable has been unplugged.

☞ *The router does not scan the configuration script for commands relating to interfaces on hot-inserted cards until the router or switch is restarted. These interfaces must be configured manually.*
The router or switch does not update the MAC address of any hot-swapped Ethernet interface until the router or switch is restarted.

The SHOW INTERFACE command is modified to show "Swapped out" in the ifOperStatus column for dormant interface instances.

All other commands that show or set board or interface properties behave as if swapped-out boards and interfaces do not exist. Commands that operate on multiple boards or interfaces skip swapped-out boards and interfaces, and commands to which a dormant interface is specified explicitly fail in their usual way for a non-existent interface.

Figure 1: Example output from the SHOW INTERFACE command.

```
Interfaces                              sysUpTime:        00:00:46

DynamicLinkTraps.....Disabled
TrapLimit............20

Number of unencrypted PPP/FR links.....0

ifIndex Interface      ifAdminStatus ifOperStatus          ifLastChange
-------------------------------------------------------------------------
   1       eth0            Up            Up                  00:00:03
   2       eth1            Up            Down                00:00:00
   3       bri0            Up            Swapped out         00:00:43
   4       eth2            Up            Swapped out         00:00:42
-------------------------------------------------------------------------


Interface name summary

Interface Full name
-------------------------------------------------------------------------
asyn0     asyn0
asyn1     asyn1
eth0      eth0
eth1      eth1
-------------------------------------------------------------------------
```

Figure 2: Example output from the SHOW INTERFACE command for a specific interface.

```
Interface................. bri0
  ifIndex................. 3
  ifMTU................... 1712
  ifSpeed................. 144000
  ifAdminStatus........... Up
  ifOperStatus............ Swapped out
  ifLinkUpDownTrapEnable... Disabled
  TrapLimit............... 20

Interface Counters

  ifInOctets ................ 52190      ifOutOctets ................ 52190
  ifInUcastPkts ............. 3070       ifOutUcastPkts ............. 3071
  ifInNUcastPkts ............ 0          ifOutNUcastPkts ............ 0
  ifInDiscards .............. 0          ifOutDiscards .............. 0
  ifInErrors ................ 0          ifOutErrors ................ 0
```

Table 1: New parameter displayed in the output of the SHOW INTERFACE command.

Parameter	Meaning
ifOperStatus	The current operational state of the interface; one of "Up", "Down", "Testing", or "Swapped Out".

Domain Name Server Enhancements

Software Release 2.3.1 includes two enhancements to Domain Name Server (DNS) functionality:

■ The router can now store recently obtained DNS information in a cache.

■ The router can now be configured to use a range of DNS servers. Server selection is based on the host name that is being resolved.

DNS Caching

DNS caching allows the router to store recently requested domain or host addresses so they can be quickly retrieved if an identical DNS request is received. DNS caching reduces traffic on the Internet and improves performance for both DNS and DNS relay under heavy usage. The DNS cache is of a limited size, and times out entries after a specified period of up to 60 minutes.

When a domain or host is requested, the cache is searched for a matching entry. If a match is found, a response is sent to the requesting PC or host. If a matching entry is not found, a request will be sent to a remote server.

First, add a DNS server to the list of DNS servers used to resolve host names into IP addresses, using the command:

```
ADD IP DNS [DOMAIN={ANY|domain-name}] {INTERFACE=interface|
    PRIMARY=ipadd [SECONDARY=ipadd]}
```

If the DNS servers have already been configured, the configuration information can be set using the command:

```
SET IP DNS [DOMAIN={ANY|domain-name}] {INTERFACE=interface|
    [PRIMARY=ipadd] [SECONDARY=ipadd]}
```

For example, to add or set the IP addresses of the default primary and secondary name servers to 192.168.20.1 and 192.168.20.2 respectively, use the commands:

```
ADD IP DNS PRIMARY=192.168.20.1 SECONDARY=192.168.20.2

SET IP DNS PRIMARY=192.168.20.1 SECONDARY=192.168.20.2
```

To set the DNS cache size and timeout values, use the command:

```
SET IP DNS CACHE [SIZE=cache-entries] [TIMEOUT=cache-max-age]
```

The name server information can be deleted from the DNS server by using the command:

```
DELETE IP DNS
```

Server Selection

The router can be configured to use a range of DNS servers with different servers being selected based on the host name being resolved.

The DOMAIN parameter in the ADD IP DNS command allows the user to specify a suffix that must be present on a host name in order for the name servers specified by the command to be used.

If the DOMAIN parameter is not specified, the name servers will be used as the default name servers. All DNS requests that do not match another specified domain will be sent to the default name servers. This is equivalent to specifying DOMAIN=ANY.

To add primary and secondary name servers with IP addresses of 202.36.163.1 and 202.36.163.3 respectively, for use as default name servers, use the command:

```
ADD IP DNS DOMAIN=ANY PRIMARY=202.36.163.1
    SECONDARY=202.36.1.3
```

These servers will be used for all host names that do not match any of the domains that are configured with their own set of name servers.

For example, to add primary and secondary name servers with IP addresses of 192.168.10.1 and 192.168.10.2 respectively, for use when resolving host names in the domain *apples.com*, use the command:

```
ADD IP DNS DOMAIN=apples.com PRIMARY=192.168.1.1
    SECONDARY=192.168.1.2
```

If a request is sent for the domain *www.fruit.apples.com*, the DNS servers at 192.168.1.1 or 192.168.1.2 will be used, as the domain matches *apples.com*.

If a request is sent for the domain *ftp.fruitpunch.apples.com*, the DNS servers at 192.168.1.1 or 192.168.1.2 will also be used, as the domain matches *apples.com*.

If a request is sent for the domain *www.armadillo.com*, the domain does not match *apples.com*, so the ANY servers 202.36.1.1 or 202.36.1.3 will be used.

Automatic Nameserver Configuration

The primary and secondary name server's addresses can either be statically configured as above, or learned dynamically over an interface. Name servers can be learned via DHCP over an Ethernet interface or via IPCP over a PPP interface. The interface is specified using the command:

```
ADD IP DNS [DOMAIN={ANY|domain-name}] INTERFACE=interface
```

If no nameservers have been manually configured, and nameserver configuration is assigned to an interface by either PPP or DHCP, this configuration will be automatically used for the default nameservers. Name servers configured in this way are identified by an "*" in the "Domain" column of the SHOW IP DNS output table. Automatically-configured nameservers can be deleted or replaced, using the commands:

```
DELETE IP DNS

SET IP DNS
```

A deleted automatic configuration may subsequently reappear if the interface concerned is reset.

Telnet Server Port Number

The listen port for the Telnet server is now configurable, so that it can be changed from the default value 23.

The LISTENPORT parameter has been added to the SET TELNET command. The syntax is:

```
SET TELNET [TERMTYPE=termstring] [INSERTNULL={ON|OFF}]
    [LISTENPORT=port]
```

The LISTENPORT parameter sets the TCP port over which the Telnet server listens for connections. If this parameter is not used, the default port number is 23.

If the TCP listen port is changed from the default of 23, care must be taken to ensure that any firewall or IP filtering configurations are matched accordingly.

Triggers for Ethernet Interfaces

Support for Ethernet UP and DOWN triggers on Ethernet interfaces has been added, as per existing triggers for other interface types. This limitation on the CREATE TRIGGER and SET TRIGGER commands is removed.

```
CREATE TRIGGER=trigger-id INTERFACE=interface EVENT={UP|DOWN|
    FAIL|ANY} [CIRCUIT=miox-circuit] [CP={APPLE|ATCP|BCP|CCP|
    DCP|DNCP|IPCP|IPXCP|LCP}] [DLCI=dlci]] [AFTER=hh:mm]
    [BEFORE=hh:mm] [{DATE=date|DAYS=day-list}] [NAME=name]
    [REPEAT={YES|NO|ONCE|FOREVER|count}] [SCRIPT=filename...]
    [STATE={ENABLED|DISABLED}] [TEST={YES|NO|ON|OFF|TRUE|
    FALSE}]
```

```
SET TRIGGER=trigger-id [INTERFACE[=interface]] EVENT={UP|
    DOWN|FAIL|ANY} [CIRCUIT=miox-circuit] [CP={APPLE|ATCP|BCP|
    CCP|DCP|DNCP|IPCP|IPXCP|LCP}] [DLCI=dlci] [AFTER=hh:mm]
    [BEFORE=hh:mm] [{DATE=date|DAYS=day-list}] [NAME=name]
    [REPEAT={YES|NO|ONCE|FOREVER|count}] [TEST={YES|NO|ON|
    OFF|TRUE|FALSE}]
```

The INTERFACE parameter defines an interface (link) trigger and specifies the interface to monitor. The EVENT parameter is required for an INTERFACE trigger. The INTERFACE parameter must be followed by the EVENT parameter. The CIRCUIT parameter may be used if INTERFACE specifies an X.25T interface; the CP parameter may be used if INTERFACE specifies a PPP interface; the DLCI parameter may be used if INTERFACE specifies a Frame Relay interface. The general trigger parameters may also be specified. The type of trigger cannot be changed.

ENCO Channels

The ENCO module provides services to user modules via channel pairs. A user module requests a service, specifying any configuration needed for the service, and is attached to an ENCO channel pair if the service and free channels are available. A channel pair consists of an encoding channel and a decoding channel. An encoding channel is used for compression, encryption, authentication or Diffie-Hellman key exchange. A decoding channel is used for decompression, decryption or authentication.

With Software Release 2.3.1, the number of channels available is now dependent on the amount of RAM on the router or switch. Routers with up to 8 MBytes of RAM (the AR300 Series) can have up to 512 encryption and compression channels. Routers with 16 MBytes (the AR700 Series) can have up to 1024 channels, and routers and switches with 32 Mbytes (the Rapier and AR800 Series) up to 2048 channels. The amount of RAM on a router or switch can be checked, using the command:

```
SHOW SYSTEM
```

The identification number of the lowest and highest channels available can be displayed, using the command:

```
SHOW ENCO
```

Information about all currently active channels, or a particular channel, can be displayed, using the command:

```
SHOW ENCO CHANNEL[=channel]
```

Note that

■   MAC cards have a limit of 128 compression channels

■   If compression is performed by the router's CPU, because a MAC card is not installed, the number of compression channels is limited, and must be configured in the boot configuration script, using the command:

```
SET ENCO SW PREDCHANNELS=0..4 STACCHANNELS=0..4
```

On AR300 Series routers the limit is two Predictor channels and four STAC LZS channels. On all other router and switch models the limit is four Predictor channels and four STAC LZS channels. By default no compression channels are configured.

IP Security (IPsec) Source Interface and Enhancements

A source interface can now be specified for tunnelled IPsec traffic. The performance of IPsec is also enhanced, and more simultaneous IPsec tunnels are supported, because of the increase in ENCO channels.

A new SRCINTERFACE parameter has been added to the SET and CREATE IPSEC POLICY commands. The SRCINTERFACE parameter specifies which interface on the router will be used as the source interface for tunnelled IPsec traffic. If the SRCINTERFACE parameter is not specified, the router defaults to the INTERFACE parameter.

The syntax for these commands is now:

```
SET IPSEC POLICY=name [ACTION={DENY|IPSEC|PERMIT}]
    [BUNDLESPECIFICATION=bundlespecification-id] [DFBIT={SET|
    COPY|CLEAR}] [GROUP={0|1|2}] [IPROUTETEMPLATE=template-
    name] [ISAKMPPOLICY=isakmp-policy-name] [LADDRESS={ANY|
    ipadd[-ipadd]}] [LMASK=ipadd] [LNAME={ANY|system-name}]
    [LPORT={ANY|OPAQUE|port}] [PEERADDRESS={ipadd|ANY|
    DYNAMIC}] [POSTION=pos] [RADDRESS={ANY|ipadd[-ipadd]}]
    [RMASK=ipadd] [RNAME={ANY|system-name}] [RPORT={ANY|port|
    OPAQUE}] [SRCINTERFACE=interface] [TRANSPORTPROTOCOL={ANY|
    EGP|ESP|GRE|ICMP|OPAQUE|OSPF|RSVP|TCP|UDP|protocol}]
    [UDPHEARTBEAT={TRUE|FALSE}] [UDPPORT=port]
    [UDPTUNNEL={TRUE|FALSE}] [USEPFSKEY={TRUE|FALSE}]

CREATE IPSEC POLICY=name INTERFACE=interface
    ACTION={DENY|IPSEC|PERMIT}
    [BUNDLESPECIFICATION=bundlespecification-id] [DFBIT={SET|
    COPY|CLEAR}] [GROUP={0|1|2}] [IPROUTETEMPLATE=template-
    name] [ISAKMPPOLICY=isakmp-policy-name]
    [KEYMANAGEMENT={ISAKMP|MANUAL}] [LADDRESS={ANY|
    ipadd[-ipadd]}] [LMASK=ipadd] [LNAME={ANY|system-name}]
    [LPORT={ANY|OPAQUE|port}] [PEERADDRESS={ipadd|ANY|
    DYNAMIC}] [POSTION=pos] [RADDRESS={ANY|ipadd[-ipadd]}]
    [RMASK=ipadd] [RNAME={ANY|system-name}] [RPORT={ANY|port|
    OPAQUE}] [SASELECTORFROMPKT={ALL|LADDRESS|LPORT|NONE|
    RADDRESS|RPORT|TRANSPORTPROTOCOL}]
    [SRCINTERFACE=interface] [TRANSPORTPROTOCOL={ANY|EGP|ESP|
    GRE|ICMP|OPAQUE|OSPF|RSVP|TCP|UDP|protocol}]
    [UDPHEARTBEAT={TRUE|FALSE}] [UDPPORT=port]
    [UDPTUNNEL={TRUE|FALSE}] [USEPFSKEY={TRUE|FALSE}]
```

where:

■ *interface* is an interface name formed by joining a layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number in the range 0 to 15 (e.g. eth0, vlan1, ppp1-1).

OSPF on Demand

OSPF on demand circuits allow data link connections to be closed when not carrying application traffic. A new parameter, DEMAND, has been added to the following commands to support this feature:

```
ADD OSPF INTERFACE [DEMAND={ON|OFF|YES|NO|TRUE|FALSE}]

SET OSPF INTERFACE [DEMAND={ON|OFF|YES|NO|TRUE|FALSE}]
```

For example, to set the OSPF interface ppp0 to a demand circuit over the point-to-point link, use the command:

```
SET OSPF INTERFACE=PPP0 DEMAND=ON
```

The DEMAND parameter specifies whether or not the interface connects to a demand circuit. Two routers connecting to the same common network segment need not agree on that segment's demand circuit status. This means that configuring one router does not require configuring other routers which connect to the same common network segment. If only one router has been configured, and the common network is a broadcast or non-broadcast multi-access (NBMA) network, the behaviour (e.g. sending, receiving hello packets) of the network will remain the same, as if the interface has not been configured as a demand circuit. If only one router has been configured and the common network segment is a point-to-point link, the router on the other end may agree to treat the link as a demand circuit and the point-to-point network receives the full benefit. When broadcast and non-broadcast multi-access (NBMA) networks are declared as demand circuits (i.e. more than one router has the network configured as a demand circuit), routing update traffic is reduced but the periodic sending of Hellos is not, which in effect still requires that the data link connection remain constantly open. The values ON, YES and TRUE are equivalent. The values OFF, NO and FALSE are equivalent. The default is OFF.

OSPF on demand is used on cost-conscious networks, such as ISDN, X.25 and dial-up networks. If there is no traffic crossing the network, (either routing protocol traffic or application traffic), the data link connection is closed. When there is traffic to send, the data link connection is established, the data is sent and the connection stays open until the link has been idle for a specified period of time. At this point the data link connection is closed to conserve cost and resources. Figure 3 and Figure 4 illustrate before and after OSPF on demand scenarios.

OSPF on demand is defined in RFC 1793, *"Extending OSPF to Support Demand Circuits"*. All routers in the network must support at least Part II of the RFC. Routers attached to on-demand links must support Part III of the RFC.

To enable OSPF to be used for routing between Router B to Router A and beyond, OSPF on demand is required. Turning OSPF on demand on at either end of the ISDN link will disable static and RIP routes.

☞ *The command ADD OSPF INTERFACE=interface VIRTUALLINK=router-id will ignore the setting of the parameter DEMAND. This means that if DEMAND is set to OFF, the virtual link is still treated as a demand circuit.*

Figure 3: Example of dial-on-demand ISDN before configuring OSPF on demand.
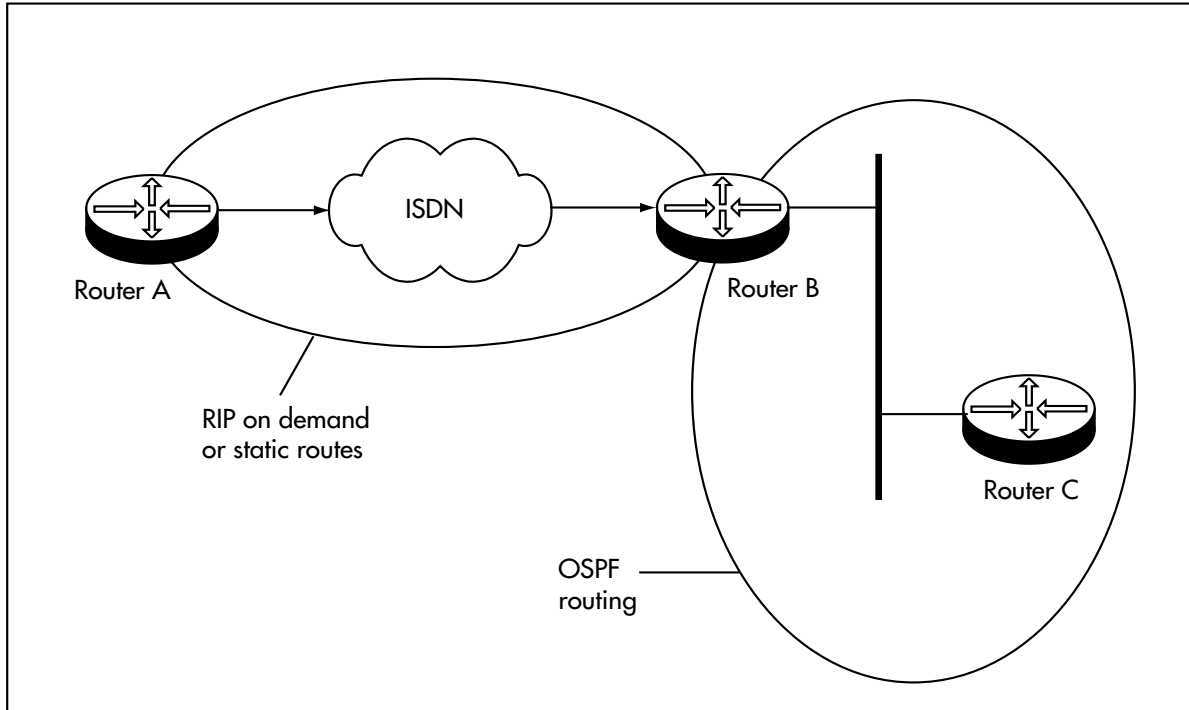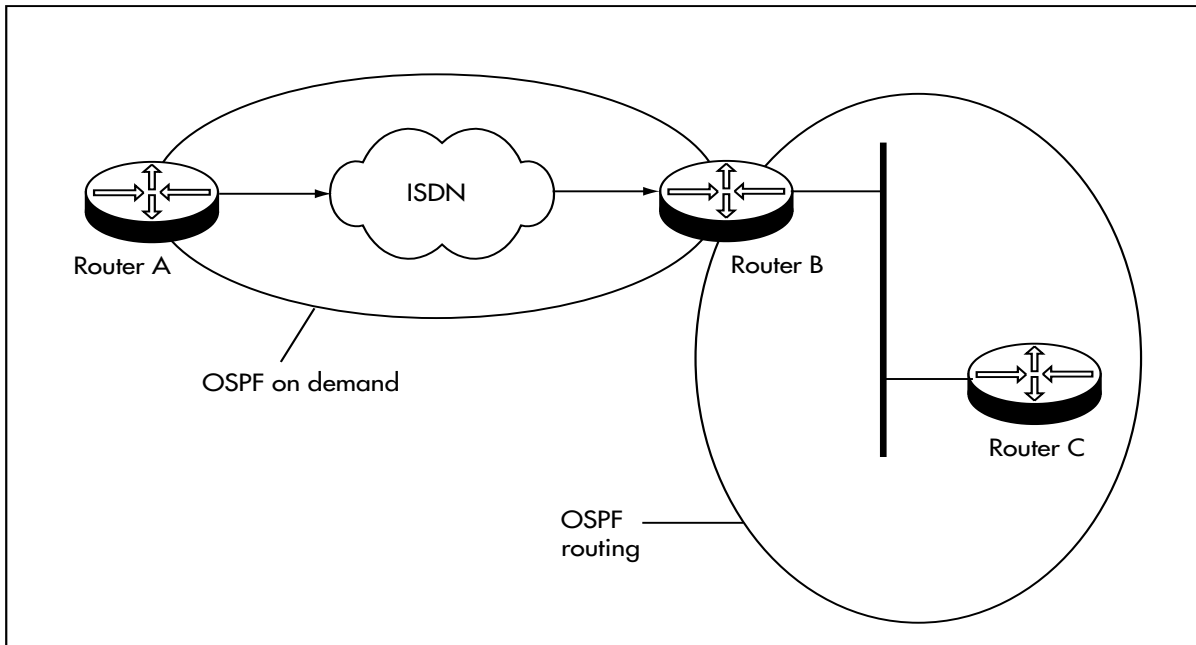

Figure 4: Example of dial-on-demand ISDN after configuring OSPF on demand.

For more information, see the *Open Shortest Path First (OSPF)* chapter of your switch or router's Software Reference. The latest Software Reference can be downloaded from the support site at *www.alliedtelesyn.co.nz/documentation/documentation.html*

Paladin Firewall Enhancements

The existing firewall NAT performs address translation for traffic passing between a pair of interfaces. With Software Release 2.3.1, firewall rules can also be configured which selectively perform address translation on sessions passing through an interface, based on the properties of the session (protocol, ports, IP addresses). In addition to standard NAT and enhanced NAT rules, it is possible to configure reverse NAT (translates destination address of outbound packets, and source address of inbound), double NAT (translates both source and destination addresses) and subnet variations of these which translate addresses from one subnet to another. Reverse enhanced NAT can also be configured, by applying an enhanced NAT rule to a public interface. Reverse enhanced NAT allows multiple inbound sessions to appear to devices on the private LAN as if all the sessions have come from the same private interface IP.

A rule can be given a limited time to live (TTL) in hours and minutes, after which it will no longer be applied and all sessions allowed by the rule will be deleted.

These features allow a service provider to bill multiple users, and provide each of them with customised, time-limited secure connections from multiple sites. For examples of their use, see "*Web Redirection with Reverse NAT Rules*" on page 18 and "*Further Examples*" on page 19.

As in previous releases, the Paladin Firewall requires a special feature licence. (Note that routers already configured to use Paladin do not require a new password.)

Interface-based NAT

The existing interface-based NAT provides a simple address translation for traffic passing between a pair of interfaces. The following methodologies are supported by interfaced-based NAT:

■   Standard NAT
     This translates the addresses of private side devices to addresses suitable for the public side of the firewall (source address will be translated for outbound packets, destination address for inbound packets).

■   Enhanced NAT
     This translates many private side addresses into a single global address suitable for use on the public side of the firewall (source address will be translated for outbound packets, destination address for inbound packets).

Rule-based NAT

The new rule-based NAT provides advanced address translation based on the properties of a packet received on a particular firewall interface. Selector values such as source address, destination address, protocol type and port number (TCP/UDP) determine which packets undergo translation. The following methodologies are supported:

■   Standard NAT
     This translates the addresses of private side devices to addresses suitable for the public side of the firewall (source address will be translated for outbound packets, destination address for inbound packets).

■ Reverse NAT
This translates the addresses of public side devices to addresses suitable for the private side of the firewall (destination address will be translated for outbound packets, source address for inbound packets).

■ Double NAT
This translates both the public and private side source and destination addresses.

■ Enhanced NAT
This translates many private or public side addresses into a single global or local address. If it is applied to a private interface the rule matches the outbound sessions (source address will be translated for outbound packets, destination address for inbound packets). If it is applied to a public interface the rule matches the inbound sessions (source address will be translated for inbound packets, destination address for outbound packets).

■ Subnet Translation
This translates IP addresses from one subnet into another subnet (e.g. all 192.168.xxx.xxx IP addresses can be translated into 202.36.xxx.xxx addresses). Subnet translation may be applied to Standard, Reverse and Double NAT.

Time Limited Rules

Rules can be set to expire after a specified Time To Live (TTL). A new parameter, TTL, specifies the time duration in hours and minutes that the rule will exist. The rule will be active from the creation of the rule and will be deleted after the time specified has expired. All entries created from this rule will be destroyed once the rule expires. Rules defined with a TTL value will not appear in router-generated configuration scripts, as they are dynamic.

New Command Syntax

The new syntax is:

```
ADD FIREWALL POLICY=policy RULE=rule-id ACTION={ALLOW|DENY|
    NAT|NONAT} INTERFACE=interface PROTOCOL={protocol|ALL|EGP|
    GRE|OSPF|SA|TCP|UDP} [AFTER=hh:mm] [BEFORE=hh:mm]
    [DAYS={MON|TUE|WED|THU|FRI|SAT|SUN|WEEKDAY|
    WEEKEND}[,...]] [ENCAPSULATION={NONE|IPSEC}] [GBLIP=ipadd]
    [GBLPORT={ALL|port[-port]}] [GBLREMOTEIP=ipadd[-ipadd]]
    [IP=ipadd[-ipadd]] [LIST={list-name|RADIUS}]
    [NATTYPE={DOUBLE|ENHANCED|REVERSE|STANDARD}]
    [NATMASK=ipadd] [PORT={ALL|port[-port]|service-name]
    [REMOTEIP=ipadd[-ipadd]] [SOURCEPORT={ALL|port[-port]}]
    [TTL=hh:mm]

SET FIREWALL POLICY=name RULE=rule-id [PROTOCOL={protocol|
    ALL|EGP|GRE|OSPF|SA|TCP|UDP}] [AFTER=hh:mm] [BEFORE=hh:mm]
    [DAYS={MON|TUE|WED|THU|FRI|SAT|SUN|WEEKDAY|
    WEEKEND}[,...]] [ENCAPSULATION={NONE|IPSEC}] [GBLIP=ipadd]
    [GBLPORT={ALL|port[-port]}] [GBLREMOTEIP=ipadd[-ipadd]]
    [IP=ipadd[-ipadd]] [NATMASK=ipadd] [PORT={ALL|port[-port]|
    service-name}] [REMOTEIP=ipadd[-ipadd]] [SOURCEPORT={ALL|
    port[-port]}] [TTL=hh:mm]
```

These commands add or modify a rule defining the access allowed between private and public interfaces of the specified policy. By default all access from public interfaces (outside the firewall) is denied and all access from private interfaces (inside the firewall) is allowed. To refine the security policy

additional rules can be added to allow or deny access based on IP addresses, port numbers, day of the week, or time of day. Each rule for a specific interface in a policy is processed in order, starting with the lowest numbered rule and proceeding to the highest numbered rule, or until a match is found.

These rules, created with the ADD FIREWALL POLICY RULE command, are based on IP address, port, protocol, date and time. In addition, the processing of ICMP packets, IP packets with options set and ping packets can be enabled or disabled on a per-policy basis using the ENABLE FIREWALL POLICY command and the DISABLE FIREWALL POLICY command.

The ACTION parameter specifies what the firewall should do with traffic that matches the selectors defined for this rule. If ALLOW is specified, the traffic will be permitted to pass through the firewall. If DENY is specified, the traffic will be prevented from passing through the firewall. If NONAT is specified, any traffic that matches the rule will not have a NAT translation performed on it, should a NAT relationship exist for the interfaces involved. If NAT is specified, the NATTYPE parameter should be used to specify whether the NAT rule performs DOUBLE, ENHANCED, REVERSE or STANDARD NAT translation. The values NONAT and NAT implicitly allow traffic through the firewall.

A rule specified with ACTION=NAT takes precedence over NAT relationships specified by the ADD FIREWALL POLICY NAT command.

A rule specified with ACTION=NAT implicitly allows traffic that matches the rule. Care should be taken when defining the rule so only the desired traffic will be permitted through the firewall.

The GBLIP parameter specifies a single IP address that is matched to the destination address of packets received on a public interface. The GBLIP parameter also specifies the global IP address to be used as the public IP address for private side devices if NAT is active on the interface, or if the value specified for the ACTION parameter is NAT.

The GBLPORT parameter specifies the port number, service name, or range of port numbers that apply to the rule if NAT is active on an interface.

The application of the GBLREMOTEIP parameter changes depending on the type of interface it is applied to. If the INTERFACE parameter specifies a public interface, it specifies a single IP address that is matched to the source IP address of packets received on that interface. If the INTERFACE parameter specifies a private interface, the GBLREMOTEIP parameter will be substituted as the destination address for packets received on the interface. This parameter should only be specified when the ACTION parameter is NAT and the NATTYPE is REVERSE or DOUBLE.

The IP parameter specifies a single IP address or a range of IP addresses that match the source address of packets received on a private interface. The IP parameter also specifies the IP address to be used as the private IP address for private side devices if NAT is active on the interface, or if the value specified for the ACTION parameter is NAT.

The NATTYPE parameter may only be used when the value specified by the ACTION parameter is NAT. It specifies whether the NAT rule performs DOUBLE, ENHANCED, REVERSE or STANDARD NAT. DOUBLE NAT

translates both the public and private side source and destination addresses. ENHANCED NAT defined for a private interface will translate the private side source address (specified using the IP parameter) and protocol dependent ports to a single source address (specified by the GBLIP parameter), suitable for the public side of the Firewall. ENHANCED NAT defined for a public interface will translate the public side source address (specified using the GBLREMOTEIP parameter) and protocol dependent ports to a single source address (specified by the REMOTEIP parameter), suitable for the private side of the Firewall. REVERSE NAT translates the addresses of public side devices (specified using the GBLREMOTEIP parameter), to addresses suitable for the private side of the Firewall (specified using the REMOTEIP parameter), so translates source address for inbound traffic and destination address for outbound traffic. STANDARD NAT translates the addresses of private side devices (specified using the IP parameter) to addresses suitable for the public side of the Firewall (specified by the GBLIP parameter), so translates source address for outbound traffic and destination address for inbound traffic.

The NATMASK parameter specifies an IP address mask that will be used to translate IP addresses from one subnet to another. The MASK parameter must only be specified when the rule action is NAT and the NATTYPE is specified as DOUBLE, REVERSE or STANDARD. The NATMASK parameter can be used when translating entire subnets from one address to another. If DOUBLE NAT is specified, the NATMASK is applied to the IP, GBLIP, REMOTEIP and GBLREMOTEIP parameters. If REVERSE NAT is specified, the NATMASK is applied to both the REMOTEIP and GBLREMOTEIP parameters. If STANDARD NAT is specified, the NATMASK is applied to both the IP and GBLIP parameters. The IP, GBLIP, REMOTEIP and GBLREMOTEIP parameters must specify a single IP address if the NATMASK parameter is used.

The REMOTEIP parameter specifies a single IP address or a range of IP addresses that match the destination address of packets received on a private interface. If the value specified for the ACTION parameter is not NAT, the REMOTEIP parameter also specifies a single IP address or range of IP addresses that match the source address of packets received on a public interface. If the value specified for the ACTION parameter is NAT, the REMOTEIP parameter also specifies the IP address to be used as the private IP address for public side devices.

Table 2 summarises the required parameters for the Firewall NAT Rules which were explained in the IP, REMOTEIP, GBLIP, GBLREMOTEIP and NATMASK paragraphs above.

Table 2: Required parameters for Firewall NAT rules.

NAT Rule Type	Direction	Parameters				
		IP	REMOTEIP	GBLIP	GBLREMOTEIP	NATMASK
Standard	I	T		S	X	X
	O			T	X	X
Standard subnet	I	T		S	X	T
	O			T	X	T
Enhanced[a]	I		T	X		X
	O			T	X	X
Reverse	I	S	T	X	S	X
	O	S	S	X	T	X
Reverse subnet	I	S	T*	X	S	T*
	O	S*	S	X	T	T*
Double	I	T	T*	S	S	X
	O	S*	S	T	T	X
Double subnet	I	T	T*	S	S	T*
	O	S*	S*	T	T	T*

a. If the rule is applied to a public interface, the result will be reverse enhanced NAT.

Key to table:

■   Direction
    I = in. The rule is applied to a public interface.
    O = out. The rule is applied to a private interface.

■   S = Selector. The value supplied for this parameter is compared to the corresponding field in a packet.

■   T = Translator. The value supplied for this parameter is substituted into the packet to bring about the address translation.

■   * = A necessary parameter. The parameter is required for the rule to function correctly, but can be put into a SET FIREWALL POLICY RULE command if the ADD command line has become too long.

■   X = Not permitted. This parameter is not permitted in this type of NAT rule.

■   Empty table entry = an optional selector.

Web Redirection with Reverse NAT Rules

The implementation of reverse NAT allows the firewall to perform Web Redirection. A NAT rule can be created which redirects HTTP traffic and sends it to one particular web server, defined in the rule, regardless of where it was originally destined. Selector parameters may also be included in the rule to fine tune which traffic is to be directed.

This feature is particularly useful for ISPs operating in the travel and hospitality industry wishing to allow users, who may previously have been unknown to the ISP, to plug their PC or laptop into the ISP's LAN. With web

redirection any web traffic from the user's PC or laptop can be redirected to the ISP's web server. This forces the user to arrange payment for using the service before being able to browse to any other site. With appropriate supporting "deny" rules, all other traffic types from the user's PC can be blocked until payment has been made.

The following gives a simple example of how a system such as this would be configured. The ISP has a switch configured with a firewall. The switch's VLANs, vlan1 and vlan2, are private and public interfaces respectively. The ISP's web server has the IP address 205.1.28.6. The following rules perform the web redirection and the blocking of all non-web traffic:

```
ADD FIREWALL POLICY=ISP RULE=298 INTERFACE=vlan1 ACTION=NAT
    NATTYPE=REVERSE PROTOCOL=TCP PORT=80 GBLREMOTE=205.1.28.6

ADD FIREWALL POLICY=ISP RULE=299 INTERFACE=vlan1 ACTION=DENY
    PROTOCOL=ALL
```

Once a user has arranged payment, a rule can be added that specifies the IP address that the ISP has assigned to the user, allowing the user full access to the service. The following is an example of such a rule. The user has been allocated the IP address 10.8.0.172. It is important that the rule number is lower than the blocking and redirecting rules, because rules are tried in order from the lowest rule number until a match is found. A low number will ensure that the allow rule will be applied if appropriate, rather than any of the other rules.

```
ADD FIREWALL POLICY=ISP RULE=5 INTERFACE=vlan1 ACTION=ALLOW
    IP=10.8.0.172 PROTOCOL=ALL
```

If the ISP wishes to take advantage of the time limited rules feature, allowing the user to have access for 30 minutes, the following rule would be used instead.

```
ADD FIREWALL POLICY=ISP RULE=5 INTERFACE=vlan1 ACTION=ALLOW
    IP=10.8.0.172 PROTOCOL=ALL TTL=0:30
```

Further Examples

Firewall and IPsec Tunnel

Enhanced NAT can facilitate routing across an IPsec tunnel, when one end of the tunnel has separate IPsec and default gateways (Figure 5 on page 20). In the following example, the router at the LAN 1 end of the tunnel has an IP address of 192.168.2.100, and the LAN 2 end of the tunnel has an IP address range of 192.168.1.1-192.168.1.100. The IP address of traffic originated by LAN 1 hosts is translated to 192.168.1.53, using the command (applied to the private eth0 interface of the LAN 1 gateway router):

```
ADD FIREWALL POLICY=zone1 RULE=7 ACTION=NAT NATTYPE=ENHANCED
    INT=eth0 PROTOCOL=all IP=192.168.2.0-192.168.2.255
    REMOTEIP=192.168.1.1-192.168.1.100 GBLIP=192.168.1.53
```

The traffic will appear to devices on LAN 2 to originate locally. When a PC in the subnet 192.168.1.1-192.168.1.100 tries to reply to a packet from a host in LAN 1 (subnet 192.168.2.0), the IPsec gateway will reply to the PC's ARP request with proxy ARP. The packet will be successfully routed through the tunnel instead of through the default gateway.

Figure 5: Using enhanced NAT in an IPsec tunnel with different IPsec and default gateways.

FW-FG1
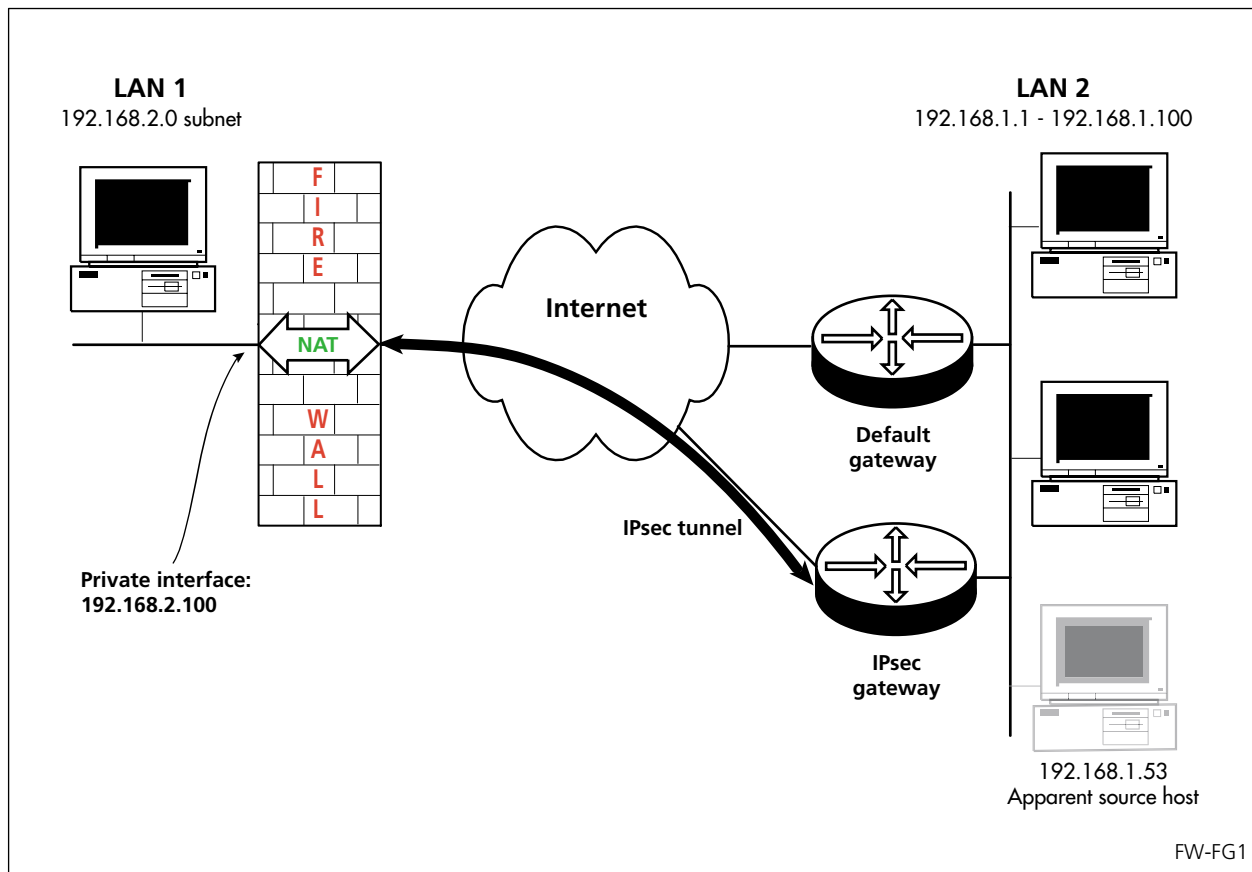
Standard NAT

To translate the source address of traffic received on the private interface *eth0* and destined for addresses in the range 210.25.4.1-210.25.4.99 to the global subnet 210.25.4.0, use the command:

```
ADD FIREWALL POLICY=zone1 RULE=10 ACTION=NAT NATTYPE=STANDARD
    INT=eth0 PROTOCOL=all GBLIP=210.25.4.0
    NATMASK=255.255.255.0 REMOTEIP=210.25.4.1-210.25.4.99
```

To provide a corresponding rule on the public interface *eth1* to translate to the private subnet 10.1.2.0, use the command:

```
ADD FIREWALL POLICY=zone1 RULE=11 ACTION=NAT NATTYPE=STANDARD
    INT=eth1 PROTOCOL=all GBLIP=210.25.4.0 IP=10.1.2.0
    NATMASK=255.255.255.0 REMOTEIP=210.25.4.1-210.25.4.99
```

Double NAT

To translate both the source and destination addresses of traffic received on the private interface with a source address of 192.168.0.74 to a destination address of 210.25.7.1 and new source address of 210.25.4.1, use the command:

```
ADD FIREWALL POLICY=zone1 RULE=50 ACTION=NAT NATTYPE=DOUBLE
    INT=eth1 PROTOCOL=all IP=192.168.0.74 GBLIP=210.25.4.1
    GBLREMOTEIP=210.25.7.1
```

Reverse NAT

To redirect all traffic received on a private interface to a destination of 210.25.7.1, without changing the source address, use the command:

```
ADD FIREWALL POLICY=zone1 RULE=51 ACTION=NAT NATTYPE=REVERSE
    INT=eth1 PROTOCOL=all GBLREMOTEIP=210.25.7.1
```

Changing Source Address

To cause all traffic that comes in over the public interface eth1 to appear to come from the private IP address 192.168.1.2, regardless of its source IP address, use the command:

```
ADD FIREWALL POLICY=zone1 RULE=60 ACTION=NAT NATTYPE=ENHANCED
    INT=eth1 PROTOCOL=all REMOTEIP=192.168.1.2
```

TTL

To modify rule number 12 in the policy named *zone3* to change the TTL value, use the command:

```
SET FIREWALL POLICY=zone3 RULE=12 TTL=1:23
```

SHOW Output

The SHOW FIREWALL POLICY and SHOW FIREWALL POLICY COUNTERS commands have been modified:

Table 3: New or modified parameters in the output of the SHOW FIREWALL POLICY command.

Parameter	Meaning
Action	The action to perform when a flow matches this rule; one of "allow", "deny", "nat" or "nonat".
NAT Type	The type of NAT translation the rule performs; one of "enhanced", "double", "reverse" or "standard".
NAT Mask	The IP address mask used to translate between subnets. Only displayed for subnet translation rules (action is "nat").

Paladin Firewall HTTP Application Gateway (Proxy)

A new Firewall HTTP proxy (Application Gateway) will filter outbound HTTP sessions based on the URLs requested, and block the setting of all cookies, or cookies requested from servers in a specified domain. The Firewall HTTP Application Gateway requires an HTTP Proxy special feature licence and an Application Gateway special feature licence, in addition to the Paladin Firewall licence.

☞ *Web browsers should not be configured to use the router or switch as a gateway or proxy for secure web traffic (HTTPS). Do not select your web browser's option for using a secure proxy or gateway, unless another device is available to provide this service.*

Firewall HTTP Proxies and Firewall Policies

To add or delete a Firewall HTTP proxy, use the new HTTP option for the PROXY parameter in the commands:

```
ADD FIREWALL POLICY=policy-name PROXY={HTTP|SMTP}
    INTERFACE=interface GBLINTERFACE=interface DIRECTION={IN|
    OUT|BOTH} [IP=ipadd] [DAYS=day-list] [AFTER=hh:mm]
    [BEFORE=hh:mm]

DELETE FIREWALL POLICY=policy-name PROXY={HTTP|SMTP}
    INTERFACE=interface GBLINTERFACE=interface DIRECTION={IN|
    OUT|BOTH} [IP=ipadd]
```

The PROXY parameter specifies the application proxy that will be added to the security policy. Available application proxies are described in Table 4.

Table 4: Application Proxies.

Proxy	Functions
HTTP	Filtering of requested URLs.
	Blocking/filtering of cookies.
SMTP	Provides filtering of spam email from known spam sources.
	Blocking of third party relay attacks.
	Blocking of email smurf amp attacks.

HTTP Filters

To add to or delete from the HTTP filter for a firewall policy, use the commands:

```
ADD FIREWALL POLICY=name HTTPFILTER=filename [DIRECTION={IN|
    OUT}]

DELETE FIREWALL POLICY=name HTTPFILTER=filename
    [DIRECTION={IN|OUT}]
```

where:

■ *name* is a character string, 1 to 15 characters in length. Valid characters are letters (a-z, A-Z), digits (0-9) and the underscore character ("_").

■ *filename* is the name of a file on the router.

These commands add or delete the contents of a HTTP filter file from the HTTP filter of the specified firewall policy. The HTTP filter file contains a list of URLs, keywords and cookie settings that are used to filter the traffic traversing the HTTP proxy.

The POLICY parameter specifies the policy to which the HTTP filter file will be added. It must already exist.

The HTTPFILTER parameter specifies the name of the HTTP filter file. The filter file is a file type with a `.txt` extension containing zero or more single line entries. The string `keywords:` must be placed at the beginning of the file and is used to start the keyword section. Keywords can be placed on the same line if they are separated by a space, or placed on separate lines. The URL section is indicated by a `URLS:` keyword as the first word on the line. URL entries must contain full domain, directory, and folder names. Only one domain is allowed

per line. Options are supplied after the entry and a colon. Each option is separated by a space.

The option keywords that are allowed for each entry are "allow" and "nocookies". The "allow" option will explicitly allow the URL, or part of the URL, given on the line. This is useful for exceptions to a deny filter or a given keyword. The "nocookies" option specifies that the proxy should not accept cookie requests from the domain or URL given, and implicitly allows the URL. Comments may be placed in the file using a # character on the beginning of the line. White space before and after an entry does not affect the parsing of the file but there must be white space between the URL and colon for the options. After the colon, white space is not needed but there must be white space between each option specified. Empty lines are also allowed. Note that all URL entries without options are considered to be denied.

How specific the URLs are determines the order of precedence of the entries in the file. For example, `www.plant.com/this/is/a/url/grow.html` would have more precedence than a entry containing `www.plant.com/this`. Also, if the allow option is specified it will have greater precedence than a similar entry with deny. Finally, keywords in the file have the least precedence. They are only applied to sections of the URL not part of the closest fitting URL entry.

Figure 6 contains an example of a URL filter file.

In order to edit the contents of the list generated from the HTTP filter file held in the firewall policy, it must be deleted from the firewall policy (using the DELETE FIREWALL POLICY HTTPFILTER command), edited and then added to the firewall policy again. Alternatively, the file may be edited. Optionally, restarting the device will reload the filter file. Editing alone does not alter the configuration held in the policy. No more than 5 URL filter files may be attached to a policy at one time.

The DIRECTION parameter specifies the direction of HTTP sessions to which the filter is to be applied. If IN is specified the filter will apply to HTTP requests that originate on the public side of the firewall (inbound). If OUT is specified the filter will apply to HTTP requests that originate on the private side of the firewall (outbound). The default value is OUT.

☞ *URL filters will have no effect unless the specified policy also has an HTTP proxy configured with a direction that matches the direction specified for the URL filter.*

For example, to add the contents of the file `banned.htp` to the HTTP filter of firewall policy zone1 for filtering outbound HTTP sessions, use the command:

```
ADD FIREWALL POLICY=zone1 HTTPFILTER=banned.htp
```

Figure 6: Example of a HTTP filter file.

```
# The keywords section starts with the string "keywords:".
keywords:
# The keywords can match any part of the URL. URLs containing these entries will
# be denied unless specifically allowed by an entry later in the file.
sex
plants
toys
.nz
# Putting a * in front of the keyword indicates that the string must appear at
# the end of the URL, for the URL to be denied. The following entry would match
www.anything.com/this/is/an/example, but not www.example.com
*example
# The * operator can be used to specify the type of file.
*.mp3
*.jpg

# The URLs section starts with the string "URLS:", and specifies particular URLs
# to deny, allow or cookie filter.
URLS:

# If no explicit deny is put on the end then the URL is denied.
# Note the implicit /* on the end of the domain.
www.plant.com
www.nude.com

# Specific sections of websites can be matched. The sections must be complete
# folder/directory names, so the following entry would match
# www.hacker.com/dosAttack/dos.html but not www.hacker.com/dosAttacks/dos.html
www.hacker.com/dosAttack

# The "nocookies" option denies cookie requests from the domain, and makes an
# implicit allow.
www.acompany.com: nocookies

# The "allow" option can be used to override general URL exclusions.
www.nude.com/this/is/not/porn : allow

# The "allow" option can also be used to override general keyword exclusions.
www.sexy.plants.com : allow

# The "allow" and "nocookies" options can be combined to allow a URL that is
# forbidden by the keywords, but deny cookie requests.
www.acompany.co.nz : allow nocookies
```

HTTP Cookies

By default, HTTP cookie requests are allowed to pass through the HTTP proxy configured under the firewall policy. To discard cookie sets from particular domains or URLs, put entries in the filter file for the direction in which you want to filter, as described above. To configure the HTTP proxy to discard all HTTP cookie sets from all responses, use the command:

```
DISABLE FIREWALL POLICY=name HTTPCOOKIES
```

where:

■   *name* is a character string, 1 to 15 characters in length. Valid characters are letters (a-z, A-Z), digits (0-9) and the underscore character ("_").

The POLICY parameter specifies the name of the firewall policy for which cookie requests are to be disabled. The policy must already exist.

To re-enable HTTP cookie requests to pass through the HTTP proxy, use the command:

```
ENABLE FIREWALL POLICY=name HTTPCOOKIES
```

For example, to enable the passing of HTTP cookies through HTTP proxies configured for the policy zone1, use the command:

```
ENABLE FIREWALL POLICY=zone1 HTTPCOOKIES
```

Firewall Policy Debugging

By default, firewall policy debugging is disabled. To enable or disable it, use the commands:

```
ENABLE FIREWALL POLICY=name [DEBUG={ALL|HTTP|PACKET|PKT|
    PROCESS|PROXY|SMTP}]

DISABLE FIREWALL POLICY=name [DEBUG={ALL|HTTP|PACKET|PKT|
    PROCESS|PROXY|SMTP}
```

The DEBUG parameter specifies the types of debugging information to be enabled. If ALL is specified, all debugging information is enabled. If HTTP is specified the display of information about request and response messages passing through the HTTP proxy is enabled. If PROXY is specified the display of general information about firewall proxies is enabled. The DEBUG parameter is not retained over a reboot.

SHOW FIREWALL POLICY

The output for the SHOW FIREWALL POLICY [COUNTER] commands include new parameters.

Table 5: New parameters in the output of the SHOW FIREWALL POLICY command.

Parameter	Meaning
HTTP Proxy Filter File	Name of a text file containing a list of domain names, keywords and cookie options that are not allowed to pass through HTTP proxies configured under this policy. This parameter is only shown if a URL filter file has been specified for this policy.
Cookies	Indicates whether or not cookies are allowed to pass through HTTP proxies configured under this policy. If "enabled" is shown all cookies are permitted unless specifically denied by an entry in the HTTP proxy filter file. If "disabled" is shown no cookies are permitted. This parameter is only shown if an HTTP proxy has been configured for this policy with direction set to "out" or "both".

Table 6: New parameters in the output of the SHOW FIREWALL POLICY COUNTER command.

Parameter	Meaning
HTTP Proxy Filter File	Name of a text file containing a list of domain names, URLs, keywords and cookie domain filters that are not allowed to pass through HTTP proxies configured under this policy. This parameter is only shown if a URL filter file has been specified for this policy.
Cookies	Indicates whether or not cookies are allowed to pass through HTTP proxies configured under this policy. If "enabled" is shown all cookies are permitted unless specifically denied by an entry in the HTTP proxy filter file. If "disabled" is shown no cookies are permitted. This parameter is only shown if an HTTP proxy has been configured for this policy with direction set to "out" or "both".
Sessions Handled	The number of TCP sessions that have been handled by the proxy.
URL Denies	The number of times a match to a requested URL has been found in the HTTP proxy filter file resulting in the request being denied.
URL Allows	The number of times a match to a requested URL has been found in the HTTP proxy filter file resulting in the request being allowed.
Cookie Denies	The number of times a match to a domain or URL requesting the setting of a cookie has been found in the HTTP proxy filter file resulting in the request being denied.

VRRP Port Monitoring

Virtual Router Redundancy Protocol (VRRP) is now able to monitor ports in the VLAN over which it is running, and reduce the priority of the router or switch if ports in the VLAN fail.

Ports that are part of a VLAN over which a VR is running can be monitored to detect port failure. This is known as *port monitoring*. Port monitoring ensures that if a port fails, or is disabled, the VRRP priority will be reduced either by a configured step value or by an amount that reflects the proportion of the VLAN's ports that are out of service. If the router is the master, and a backup router has a higher priority, the backup router pre-empts the master and becomes the new master.

Port monitoring is a way of implementing a connectivity metric. If the connectivity to the VLAN changes, the router will drop its priority either proportionally or by a certain amount by using the STEPVALUE parameter of the following command:

```
SET VRRP=vr-identifier [PORTMONITORING={ON|OFF}]
    [STEPVALUE={stepvalue|PROPORTIONAL}]
```

If the *stepvalue* option is specified, the priority of the VR will be reduced by this value each time a VLAN port fails or is disabled.

If the PROPORTIONAL option is specified, the virtual router reduces the priority to a percentage of the original priority in proportion the percentage of available ports. For example, if a router has five ports and a port fails, the router will drop its priority by a fifth of the original priority.

```
SET VRRP=vr-identifier [ADINTERVAL=1..255]
   [AUTHENTICATION={NONE|PLAINTEXT}] [PASSWORD=password]
   [PORTMONITORING={ON|OFF} [STEPVALUE={stepvalue|
   PROPORTIONAL}]] [PREEMPT={ON|OFF}] [PRIORITY=1..254]
```

where:

■ *stepvalue* is a decimal number in the range 1 to 254.

This command only changes the parameters on this router. It is important that all routers involved in a virtual router are configured with the same values for the VRRP virtual router identifier, IP address, advertisement interval, pre-empt mode, authentication type, and password. Inconsistent configuration will cause advertisement packets to be rejected and the virtual router will not perform properly.

The PORTMONITORING parameter is valid when the VR is providing redundancy over a VLAN. The PORTMONITORING parameter specifies whether the VRRP should monitor the ports of the VLAN and alter the priority value if ports fail or are disabled. If the PORTMONITORING parameter is set to ON, the STEPVALUE parameter may also be specified. The default is OFF.

The STEPVALUE parameter specifies the value by which the priority of the VR should be decremented each time a VLAN port fails, or is disabled when the PORTMONITORING parameter is set to ON. If a number is specified, the priority of the VR will be reduced by this value each time a VLAN port fails or is disabled. If PROPORTIONAL is specified, the VR will reduce the priority to a percentage of the original priority in proportion to the percentage of available ports. The value specified for the STEPVALUE parameter is retained when port monitoring is disabled.

For instance, to enable the PORTMONITORING feature on the virtual router number 10 and set the step value to 100, use the following command:

```
SET VRRP VRID=10 PORTMONITORING=ON STEPVALUE=100
```

The SHOW VRRP command now shows port monitoring information.

Table 7: New parameters displayed in the output of the SHOW VRRP command.

Parameter	Meaning
Port Monitoring	Indicates whether the port monitoring feature is ON or OFF. This parameter is only displayed if the VR is operating over a VLAN interface.
Step value	If a number is shown (e.g. "40"), this indicates the value by which the priority of the VR will be reduced for each VLAN port that fails or is disabled. If "PROPORTIONAL" is shown, the priority will be reduced in proportion to the percentage of VLAN ports that are out of service.

Border Gateway Protocol 4 (BGP-4)

The *Border Gateway Protocol version 4* (BGP-4) is an external gateway protocol which allows two routers in different routing domains to exchange routing information. Software release 2.3.1 supports phase one implementation of BGPv4 on AR700 Series routers, Rapier Series Layer 3 Switches, and AR800 Series Modular Switching Routers. BGP-4 is not available on AR300 Series routers.

The available commands are:

- ADD BGP AGGREGATE

- ADD BGP CONFEDERATIONPEER

- ADD BGP IMPORT

- ADD BGP NETWORK

- ADD BGP PEER

- DELETE BGP AGGREGATE

- DELETE BGP CONFEDERATIONPEER

- DELETE BGP IMPORT

- DELETE BGP NETWORK

- DELETE BGP PEER

- DISABLE BGP DEBUG

- DISABLE BGP PEER

- ENABLE BGP DEBUG

- ENABLE BGP PEER

- RESET BGP PEER

- SET BGP

- SET BGP AGGREGATE

- SET BGP IMPORT

- SET BGP PEER

- SHOW BGP

- SHOW BGP AGGREGATE

- SHOW BGP CONFEDERATION

- SHOW BGP IMPORT

- SHOW BGP NETWORK

- SHOW BGP PEER

- SHOW BGP ROUTE

For more information, see the *Border Gateway Protocol version 4 (BGP-4)* chapter of your switch or router's Software Reference. The latest Software Reference can be downloaded from the support site at *www.alliedtelesyn.co.nz/documentation/documentation.html*

Internet Protocol (IP)

In conjunction with BGP-4, a number of new commands have been added to the implementation of IP, and several commands have been modified.

The new commands are:

■ ADD IP ASPATHLIST

■ ADD IP COMMUNITYLIST

■ ADD IP ROUTEMAP

■ DELETE IP ASPATHLIST

■ DELETE IP COMMUNITYLIST

■ DELETE IP ROUTEMAP

■ SET IP AUTONOMOUS

■ SET IP ROUTEMAP

■ SHOW IP ASPATHLIST

■ SHOW IP COMMUNITYLIST

■ SHOW IP ROUTEMAP

The modified commands are:

■ ADD IP FILTER

■ DELETE IP FILTER

■ SET IP FILTER

■ SHOW IP FILTER

For more information, see the *Internet Protocol (IP)* chapter of your switch or router's Software Reference. The latest Software Reference can be downloaded from the support site at *www.alliedtelesyn.co.nz/documentation/ documentation.html*

IP and Interface Counters

Software Release 2.3.1 allows you to reset all MIB counters to zero for a specified interface, or for all interfaces, or to reset all or individual IP counters.

To reset counters based on interface, use the command:

```
RESET INTERFACE[={ifIndex|interface}] COUNTER
```

where:

■ *ifIndex* is a decimal value specifying the entry in the interface MIB.

■ *interface* is an interface name formed by concatenating an interface type and number.

This command resets the MIB counters for the interfaces specified. If an interface is not specified, all the MIB counters will be reset. For example, to reset the eth0 interface MIB counters, use either of the following commands:

```
RESET INTERFACE=eth0 COUNTER

RESET INTERFACE=1 COUNTER
```

To reset IP interfaces, use the command:

```
RESET IP COUNTER={ALL|ARP|EGP|ICMP|INTERFACE|IP|MULTICAST|
     ROUTE|SNMP|UDP}
```

This command resets the specified group of IP counters to zero (0). The COUNTER parameter specifies the group of counters to be reset. If ALL is specified, all IP counters are reset. If ARP, EGP, ICMP, INTERFACE, IP, MULTICAST, ROUTE, SNMP or UDP is specified then those counters, respectively, are reset. For example, to reset the IP route counters to zero, use the command:

```
RESET IP COUNTER=ROUTE
```

SNMP and ARP counters are now displayed with the SHOW IP COUNTER command. RIP is no longer a valid option for the SHOW IP COUNTER command; to display RIP counters, use the command:

```
SHOW IP RIP COUNTER
```

The new syntax for the SHOW IP COUNTER command is:

```
SHOW IP COUNTER[={ALL|ARP|EGP|ICMP|INTERFACE|IP|MULTICAST|
     ROUTES|SNMP|UDP}]
```

This command displays all or selected parts of the IP MIB. If an option is not specified, or ALL is specified, all the IP counters are displayed. The output displayed with the option ARP is shown in Figure 7 on page 30, and Table 8 on page 30. The output displayed with the option INTERFACE has changed, and is shown in Figure 8 on page 31 and Table 9 on page 31. The output displayed with the option SNMP is shown in Figure 9 on page 32 and Table 10 on page 32.

Figure 7: Example output from the SHOW IP COUNTER=ARP command.

```
ARP counter

  arpRxPkts ........................... 0    arpTxPkts ........................... 0
  arpRxReqPkts ........................ 0    arpTxReqPkts ........................ 0
  arpRxRespPkts ....................... 0    arpTxRespPkts ....................... 0
  arpRxDiscPkts ....................... 0    arpTxDiscPkts ....................... 0
```

Table 8: Parameters displayed in the output of the SHOW IP COUNTER=ARP command.

Parameter	Meaning
arpRxPkts	The number of ARP packets received.
arpRxReqPkts	The number of ARP Request packets received.
arpRxRespPkts	The number of ARP Response packets received.
arpRxDiscPkts	The number of inbound ARP packets discarded.
arpTxPkts	The number of ARP packets transmitted.
arpTxReqPkts	The number of ARP Request packets transmitted.
arpTxRespPkts	The number of ARP Response packets transmitted.
arpTxDiscPkts	The number of outbound ARP packets discarded.

Figure 8: Example output from the SHOW IP COUNTER=INTERFACE command.

```
IP Interface Counters
--------------------------------------------------------------------------------
Interface       ifInPkts     ifInBcastPkts    ifInUcastPkts     ifInDiscards
Type            ifOutPkts    ifOutBcastPkts   ifOutUcastPkts    ifOutDiscards
--------------------------------------------------------------------------------
eth0              23531           23224            307                 0
Static              230               0            230                 0

eth1                  0               0              0                 0
Static            63289           63289              0                 0

ppp0                  0               0              0                 0
Static                0               0              0                 0
--------------------------------------------------------------------------------
```

Table 9: Parameters displayed in the output of the SHOW IP COUNTER=INTERFACE command.

Parameter	Meaning
Interface	The name of the interface (e.g. ETH0, PPP0, FR0), or "LOCAL" for the local IP interface. If multihoming is enabled (two or more logical interfaces have been assigned to a single layer 2 interface), all interface names will include a hyphen ("-") and the logical interface number.
Type	The type of interface; one of "Static", "Dynamic" or "Inactive". A static interface is a permanent interface that is active and in use. A dynamic interface is a non-permanent interface created by Asynchronous Call Control (ACC) when a dial-in user initiates a SLIP or PPP connection. The interface will disappear when the user logs off, when the router is restarted or when the IP module is reset with the RESET IP command. An inactive interface is a permanent interface that could not attach to the lower-layer (FR, PPP, ETH, etc) interface for some reason. The interface is not in use but remains configured and will become active if the lower-layer attachment succeeds on the next RESET IP or restart. The most common cause of inactive interfaces is the deletion of the lower-layer interface. Inactive interfaces may be deleted by the manager, but can not be modified.
ifInPkts	The number of packets received via the interface.
ifOutPkts	The number of packets transmitted via the interface.
ifInBcastPkts	The number of multicast packets received via the interface.
ifOutBcastPkts	The number of multicast packets transmitted via the interface.
ifInUcastPkts	The number of unicast packets received via the interface.
ifOutUcastPkts	The number of unicast packets transmitted via the interface.
ifInDiscards	The number of packets received via the interface that were discarded.
ifOutDiscards	The number of packets to be transmitted via the interface that were discarded.

Figure 9: Example output from the SHOW IP COUNTER=SNMP command.

```
SNMP counters:
  inPkts ......................... 0          outPkts ........................ 0
  inBadVersions .................. 0          outTooBigs ..................... 0
  inBadCommunityNames ............ 0          outNoSuchNames ................. 0
  inBadCommunityUses ............. 0          outBadValues ................... 0
  inASNParseErrs ................. 0          outGenErrs ..................... 0
  inTooBigs ...................... 0          outGetRequests ................. 0
  inNoSuchNames .................. 0          outGetNexts .................... 0
  inBadValues .................... 0          outSetRequests ................. 0
  inReadOnlys .................... 0          outGetResponses ................ 0
  inGenErrs ...................... 0          outTraps ....................... 0
  inTotalReqVars ................. 0
  inTotalSetVars ................. 0
  inGetRequests .................. 0
  inGetNexts ..................... 0
  inSetRequests .................. 0
  inGetResponses ................. 0
  inTraps ........................ 0
```

Table 10: Parameters in the output of the SHOW IP COUNTER=SNMP command.

Parameter	Meaning
inPkts	The number of SNMP packets received by the router.
inBadVersions	The number of SNMP packets with a bad version field received by the router.
inBadCommunityNames	The total number of SNMP PDUs delivered to the SNMP agent that used an unknown SNMP community name.
inBadCommunityUses	The total number of SNMP PDUs delivered to the SNMP agent that represented an SNMP operation not allowed by the SNMP community name in the PDU.
inASNParseErrs	The total number of ASN.1 parsing errors, either in encoding or syntax, encountered by the SNMP agent when decoding received SNMP PDUs.
inTooBigs	The total number of valid SNMP PDUs delivered to the SNMP agent for which the value of the errorStatus component was tooBig.
inNoSuchNames	The number of SNMP packets received with an error status of nosuchname.
inBadValues	The number of SNMP packets received with an error status of badvalue.
inReadOnlys	The number of SNMP packets received with an error status of readonly.
inGenErrs	The number of SNMP packets received with an error status of generr
inTotalReqVars	The total number of SNMP MIB objects requested.
inTotalSetVars	The total number of SNMP MIB objects which were changed.
inGetRequests	The number of SNMP Get Request packets received by the router.
inGetNexts	The number of SNMP Get Next Request packets received by the router.

Table 10: Parameters in the output of the SHOW IP COUNTER=SNMP command.

Parameter	Meaning
inSetRequests	The number of SNMP Set Request packets received by the router.
inGetResponses	The number of SNMP Get Response packets received by the router.
inTraps	The number of SNMP trap message packets received by the router.
outPkts	The number of SNMP packets transmitted by the router.
outTooBigs	The number of SNMP packets transmitted with an error status of toobig.
outNoSuchNames	The number of SNMP packets transmitted with an error status of nosuchname.
outBadValues	The number of SNMP packets transmitted with an error status of badvalue.
outGenErrs	The number of SNMP packets transmitted with an error status of generror.
outGetRequests	The number of SNMP Get Request response packets transmitted by the router.
outGetNexts	The number of Get Next response packets transmitted by the router.
outSetRequests	The number of Set Request packets transmitted by the router.
outGetResponses	The number of SNMP Get response packets transmitted.
outTraps	The number of SNMP Traps transmitted by the router.

Telephony (PBX) Functionality

AR300 Series routers with telephony ports now offer a choice of ISDN supplemental services or internal PBX functions. The PBX functions are enabled by default, but one or more extensions can be set to support ISDN supplemental services instead, using the commands:

```
CREATE PBX EXTENSION=extension_number RECALL={LOCAL|REMOTE}

SET PBX EXTENSION=extension_number RECALL={LOCAL|REMOTE}
```

If LOCAL is specified, PBX functions are enabled. If REMOTE is specified, ISDN supplemental services are enabled.

For more information, see the *Telephony Services* chapter of your router's Software Reference. The latest Software Reference can be downloaded from the support site at *www.alliedtelesyn.co.nz/documentation/documentation.html*

Bandwidth Limiting

This feature will be available on Rapier i Series layer 3 switches only, when these models become available.

Ingress and egress bandwidth limits are specified separately. Limits can be configured for each switch port using the command:

```
SET SWITCH PORT=port-list
    [INGRESSLIMIT={NONE|DEFAULT|0|64..127000|8..1016}]
    [EGRESSLIMIT={NONE|DEFAULT|0|1000..127000|8..1016}]
```

where:

■   *port-list* is a port number, a range of port numbers (specified as n-m), or a comma separated list of port numbers and/or ranges. Port numbers start at 1 and end at m, where m is the highest numbered Ethernet switch port, including uplink ports.

The INGRESSLIMIT parameter specifies the maximum bandwidth for traffic ingressing the specified port(s), in kbps (10/100 Mbps ports) or Mbps (Gigabit ports). If NONE or 0 (zero) is specified, ingress limiting is disabled for the specified port. For 10/100 Mbps ports the input value (64..127000) in kbps is rounded down to the nearest 64kbps if below 1000, otherwise it is rounded down to the nearest 1000 (or 1 Mbps). For Gigabit ports the input value (8..1016) in Mbps is rounded down to the nearest 8 Mbps. The default is NONE.

The EGRESSLIMIT parameter specifies the maximum bandwidth for traffic egressing the specified port(s), in kbps (10/100 Mbps ports) or Mbps (Gigabit ports). If NONE or 0 (zero) is specified, egress limiting is disabled for the specified port. For 10/100 Mbps ports the input value (1000..127000) in kbps is rounded down to the nearest 1000 (or 1 Mbps). For Gigabit ports the input value (8..1016) in Mbps is rounded down to the nearest 8 Mbps. The default is NONE.

Errata: Telnet Server

The following commands can be used to disable or enable the Telnet Server, and to show information about its current settings. These commands apply to current and previous releases, and will appear in the next revision of the Software Reference for all models.

DISABLE TELNET SERVER

Syntax   `DISABLE TELNET SERVER`

Description   This command disables the Telnet server from being accessed remotely. The Telnet server is enabled by default.

ENABLE TELNET SERVER

Syntax   `ENABLE TELNET SERVER`

Description   This command enables the Telnet server to be accessed remotely. The Telnet server is enabled by default.

SHOW TELNET

Syntax   `SHOW TELNET`

Description   This command displays information about the current Telnet settings.

Figure 10: Example output from the SHOW TTY command.

```
TELNET Module Configuration
------------------------------------------------------------------------------
  Telnet Server .................. Enabled
  Telnet Server Listen Port ...... 23
  Telnet Terminal Type ........... UNKNOWN
  Telnet Insert Nulls ............ Off
------------------------------------------------------------------------------
```

Table 11: Parameters displayed in the output of the SHOW TELNET command.

Parameter	Meaning
Telnet Server	The status of the Telnet Server, either "Enabled" or "Disabled".
Telnet Server Listen Port	The TCP port number that the Telnet server is listening on. Can be any number from 1 to 65535 that is not already in use.
Telnet Terminal Type	The terminal type identification string that is passed to a remote Telnet server upon connection. The default is "UNKNOWN".
Telnet Insert Nulls	Determines if a NULL character should be inserted after each CR that is sent, either "On" or "Off".

Installation

There are no issues upgrading from Software Releases 2.2.*x* to Software Release 2.3.1.

Software Release 2.2.1 added a new parameter, LOGIN, to the ADD USER and SET USER commands:

```
ADD USER=login-name PASSWORD=password LOGIN={TRUE|FALSE|ON|
    OFF|YES|NO} [other-options...]

SET USER=login-name LOGIN={TRUE|FALSE|ON|OFF|YES|NO}
    [other-options...]

SET USER LOGIN={TRUE|FALSE|ON|OFF|YES|NO} [other-options...]
```

The LOGIN parameter is used to specify whether or not users with a privilege of "user" will be able to login to the command line interface. Usernames with LOGIN set to TRUE can be used both for PAP and CHAP authentication, and to login and access the command line. Usernames with LOGIN set to FALSE can only be used for PAP and CHAP authentication.

After upgrading from 2.0.x or 2.1.x to 2.3.x, the LOGIN parameter is required. If it is not set in a boot script, its value defaults to TRUE, allowing users with a privilege level of "user" to access the CLI. The manager then has the option of denying access to some or all users.